

TUGAS AI k-Nearest Neighbour

1. Study Kasus

Dalam tugas ini, yang menjadi tugas *program* adalah *learning* dengan algoritma k-Nearest Neighbour (k-NN) pada *dataset* dengan tujuan klasifikasi *label* pada *data testing*.

Untuk melakukan k-Nearest Neighbour dibutuhkan nilai K optimal (*bestK*) dengan mekanisme k- Fold Cross Validation kemudian k tersebut digunakan untuk melakukan klasifikasi pada *data testing* dengan kNN. Setelah itu, prediksi yang didapat akan disimpan kedalam file csv.

2. Analisis dan pembahasan

a. k-Nearest Neighbour

k-Nearest Neighbour adalah salah satu metode klasifikasi pada *machine learning* dan juga kNN yang paling mudah dimengerti (*straightforward*) karena proses klasifikasi pada metode ini dengan mengidentifikasi sebuah *data testing* dengan tetangga terdekat dari *data training* dan menggunakan tetangga tersebut untuk memberikan data testing tersebut sebuah class berdasarkan class tetangga. Untuk menghitung jarak tetangga terdekat, metode yang digunakan adalah *Euclidean Distance*

Pseudocode k-NN sebagai berikut :

Gambar 1. Pseudocode KNN

Langkah-langkah untuk menghitung metode *K-Nearest Neighbor* [6] antara lain :

1. Menentukan parameter *K* (jumlah tetangga paling dekat).
2. Menghitung kuadrat jarak *Euclid* (*query instance*) masing-masing objek terhadap data sampel yang diberikan menggunakan persamaan 1.
3. Kemudian mengurutkan objek-objek tersebut ke dalam kelompok yang mempunyai jarak *Euclid* terkecil.
4. Mengumpulkan kategori *Y* (Klasifikasi *Nearest Neighbor*)
5. Dengan menggunakan kategori *Nearest Neighbor* yang paling mayoritas maka dapat diprediksi nilai *query instance* yang telah dihitung.

b. k Fold Cross Validation

K-fold cross validation merupakan salah satu metode yang digunakan untuk mengetahui rata-rata keberhasilan dari suatu sistem dengan cara melakukan perulangan dengan mengacak atribut masukan sehingga sistem tersebut teruji untuk beberapa atribut input yang acak. K-foldcross validation diawali dengan membagi data sejumlah *nfold* yang diinginkan.

langkah-langkah yang dilakukan dalam membangun program adalah sebagai berikut :

1. Mengimport dataset ke variable yang ada
2. Melakukan k-Fold Cross Validation pada *dataTraining* maksimal untuk kNN adalah 500 dengan pertambahan k adalah bilangan ganjil, dan akurasi masing-masing k.

3. Mencari bestK dari hasil k-Fold Cross Validation dengan pengujian nilai k yang bernilai ganjil.
4. Menggunakan k untuk dilakukan klasifikasi KNN untuk mengklasifikasikan data Testing dan menghasilkan output bestK, visualisasi k dan akurasi yang didapat dari *confusion matrix*, serta hasil klasifikasi.
5. Membuat file excel yang berisi label dari klasifikasi.

3. Hasil

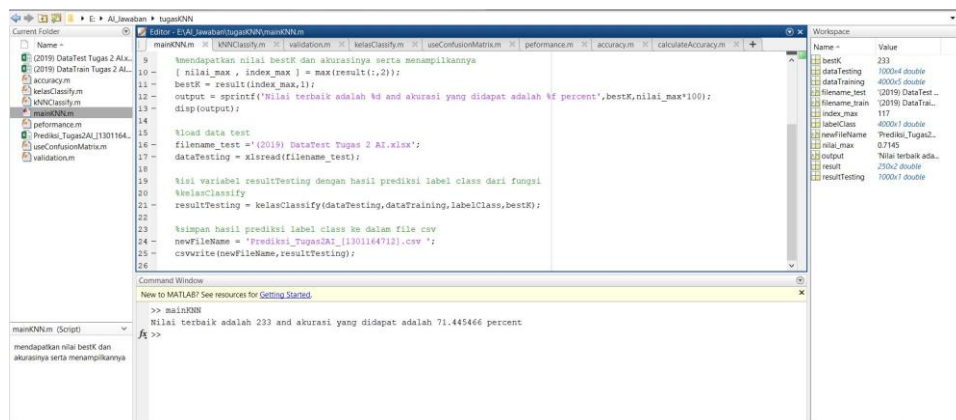
Tabel 1. Pengujian

No	Percobaan	Batas nilai k	Nilai bestK	Akurasi(%)
1	Percobaan ke-1	50	49	70.332795 %
2	Percobaan ke-2	200	189	71.366740 %
3	Percobaan ke-3	500	233	71.445466 %

Berdasarkan tabel diatas dapat disimpulkan bahwa parameter *bestK* merupakan salah satu parameter yang paling optimum pada kasus/problem yang dihadapi karena nilai k dapat meningkatkan maupun menurunkan akurasi prediksi.

Berikut *screenshot* visualisasi plot akurasi sesuai nilai k serta *screenshot* hasil *running* terbaik berdasarkan nilai-nilai parameter optimum tersebut

Gambar 2. *Screenshot* hasil *running* terbaik



Gambar 3. Plot akurasi sesuai nilai k

